SYSTEM AND METHOD FOR RELIABLY PURGING STATISTICAL RECORDS

I.   Background

A. Field of the Invention

[001]   This invention relates generally to the field of network management, and more particularly to maintenance operations on elements within a managed telecommunications network.

B. Copyright Notice/Permission

[002]   A portion of the disclosure of this patent document contains material that is subject to copyright protection. The copyright owner has no objection to the reproduction by anyone of the patent document or the patent disclosure as it appears in the Patent and Trademark Office patent file or records, but otherwise reserves all copyright rights whatsoever. The following notice applies to the software and data as described below and in the drawings hereto: Copyright. COPYRGT.2001-002, BellSouth Intellectual Property Management Corporation.

C. Description of the Related Art

[003]   Telecommunications companies (i.e., service providers) build, operate, and maintain very large communications and related networks. Part of the operation and maintenance of these networks involves the use of operations software, typically divided into a number of functional areas such as engineering, provisioning, and the like. Provisioning software aids service providers in receiving requests for service or alterations to existing service, be it voice and/or data, and configuring both the telecommunications network and/or related networks and systems (e.g., accounting, billing, and the like) to provide the new service requested. Engineering operations software in contrast is typically used by service providers to configure and monitor network elements to ensure they perform their functions properly. Service providers also use engineering operations software to facilitate service provisioning and monitoring.

One of the primary engineering operations software systems is the element management system (EMS) software. Typical EMS packages are centralized service network management applications that manage and control (typically via standards such as SNMP and the like) the various elements in the telecommunications and/or related networks. Within the core telecommunications network the elements often are

multiservice elements such as frame relay, SMDS, ATM, IP, and/or the like switches. Some of the operations performed by typical EMS packages include: circuit provisioning to establish end-to-end network connectivity; logical provisioning of individual circuits and to establish network-wide parameters; providing audit trails on activities such as the length of a user session and the addition or modification of switches, logical ports, trunks, circuits, and the like; display of network statistics for real-time status information on logical and physical ports; display of usage data on logical and physical ports and the like for network planning and trend analysis; and collecting different types of traps for alarm indications and statistics logging for the numerous objects in the telecommunications networks (e.g., switches, trunks, physical ports, logical ports, permanent virtual circuits, switched virtual circuits, and the like).

[004]   With regard to gathering statistics in particular, the EMS package typically collects records from the various elements in the network being managed and sends them to a central repository comprised of one or more statistics databases via one or more statistics servers. However, with the explosive growth in demand for telecommunications services over the past few years the number of elements within the service providers' networks have dramatically increased. As a result, the number of statistic records generated by the various elements in service providers' networks has swelled, thereby generating so many records at a such a rapid pace that existing systems and methods of collecting, analyzing, and managing these statistic records often have been overwhelmed. Accordingly, there is a need for improved systems and methods of collecting and managing statistical records in telecommunications and/or related networks.

II.   Summary of the Invention

[005]   In a telecommunications system having a plurality of managed elements, many if not all of the managed elements generating statistical data that is communicated to one or more statistics servers, an improved statistical record purge procedure, the improvement comprising running the purge procedure less often and at times when there is likely to be relatively little new statistical data generated by the managed elements. Existing purge procedures also may be improved by ensuring there is adequate temporary storage or log space in the one or more statistics servers for new statistical records generated by the statistics servers while the corresponding statistics database is having older records expunged. The purge procedure also may be improved through the inclusion

of instructions that update or reset any pointers or indices in the statistics database after older records have been expunged.

III.   Brief Description of the Drawings

[006]   These and other features, aspects, and advantages of the invention will become better understood in connection with the appended claims and the following description and drawings of various embodiments of the invention where:

Fig. 1 illustrates and an exemplary network within which the invention may be implemented;

Fig. 2 illustrates the structure of an exemplary server that may reside within a network such as that illustrated in Fig. 1; and

Fig. 3 illustrates in flow diagram form a method for improving existing statistical record purge procedures in accordance with an exemplary embodiment of the invention.

IV.   Detailed Description of the Preferred Embodiments

[007]   Throughout the following detailed description similar reference numbers refer to similar elements in all the figures of the drawings.

[008]   Fig. 1 illustrates an exemplary network 101 in which the invention may be implemented. Network 101 is based in part on the EMS developed and marketed by Lucent Technologies of Murray Hill, New Jersey under the trademark NAVISCORE. The NAVISCORE EMS is a distributed multiservice element manager that utilizes a graphically integrated UNIX-based platform and telecommunications network management (TNM) standards to perform its network management and control functions. Network 101 also includes portions of a suite of management servers developed and marketed by Lucent Technologies under the trademark NAVISEXTEND ENVIRONMENT. The NAVISEXTEND ENVIRONMENT extends the functionality of the NAVISCORE EMS. Network 101 as depicted includes a plurality of fault servers 102 and statistics servers 103 operatively connected to a private network 104. Network 101 also includes a fault database 105 and a statistics database 106 operatively connected to private network 104. As will be understood by one skilled in the art, network 101 need not include many of the elements depicted therein (e.g., fault servers 102, firewalls, DMZ network 108, and the like), and may include any number of other elements not depicted in Fig. 1 (e.g., provisioning servers, accounting servers, and the like).

In operation, the various switches and/or other managed network elements (not shown) in the telecommunications network 107 compile statistical data about their operations almost constantly. Periodically the statistics servers 103 poll the various switches and/or other elements in network 107 and the switches and/or other elements send to the statistics servers (via the demilitarized zone (DMZ) network and the private network 104) the statistical data each has compiled since it was last polled. The statistics servers 103 convert the statistical data into American Standard Code for Information Interchange (ASCII) formatted records and store the ASCII formatted messages in temporary memory or log space. The ASCII formatted statistical records are then sent periodically by the statistics servers 103 to the statistics database 106 via the private network 104, where the statistical records are stored in permanent memory and may be accessed by other systems in the network for analysis, troubleshooting, and the like. For example, the statistical records stored in statistics database 106 may be accessed by a report generator or similar software package (not shown) residing on another server such as a report server (not shown) so that real-time or close to real-time statistical data may be accessed by network operators.

[009]   While one skilled in the art will understand that servers 103 may be implemented in any number configurations on any number of computing platforms, Fig. 2 illustrates a generic computing platform 201 for servers 103. As shown, computing platform 201 includes processing unit 222, system memory 224, and system bus 226 that couples various system components including system memory 224 to the processing unit 222. The system memory 224 might include read-only memory (ROM) and/or random access memory (RAM). The platform 201 might further include a hard-drive 228, which provides storage for computer readable instructions, data structures, program modules, other data, and the like. A user may enter commands and information into the platform 201 through input devices such as a keyboard 240 and pointing device 242. A monitor 244 or other type of display device may also be connected to the platform 201 for visual output. Communications device 243, which may be for example a TCP/IP enabled device, provides for connectivity to other computing devices within or beyond network 101 illustrated in Fig. 1. Processor 222 may be programmed with instructions to interact with other computing systems so as to perform the algorithms and operations described below. Processor 222 may be loaded with any one of several computer operating systems such as Windows NT, Windows 2000, Linux, and the like. In a particular embodiment of the

4

invention, processing unit 222 comprises a 4x450 MHz CPU, system memory 224 comprises 4 Gigabytes of RAM, hard-drive 228 comprises a 36 Gigabyte disk-drive, and processor 222 includes a UNIX segment.

[0010]   Because the information contained in the statistical records stored in permanent memory becomes stale at some point and the amount of storage space in the statistics database 106 or other permanent memory is necessarily limited, a purge script is run periodically to expunge a predetermined number of older statistics records stored in the statistics database 106. In one configuration of the statistics servers 103 a scheduler initiates the purge procedure, and a purge script ultimately calls on a Sybase stored procedure that resides in a UNIX-based segment of statistics database 106. Optimally, older statistical records would be kept for the duration of their usefulness while no younger or fresher statistical records would be lost due to insufficient storage space in the statistics database 106 or other permanent memory. In current networks it is not unusual to run the purge procedure numerous times per hour due to the sheer number of statistical records being produced by the large numbers of switches and/or other elements in the service providers' networks.

[0011]   In the networks owned and operated by the assignee of the present invention however, running the statistical record purge procedure up to several times per hour, or even hourly, resulted in the loss of new statistical records (i.e., records generated during the period the purge procedure was running). We determined these losses were due in part to there being inadequate log space in the statistics server to temporarily store statistical records while the purge procedure removed old records from the statistics database 106. We also determined the foregoing losses were due to statistics database 106 deadlock errors occurring when the processes within the purge procedure and other processes running on the statistics server tried to access the statistics database simultaneously.

[0012]   We therefore developed the following improvements to existing statistical record purge procedures to address the problems outlined above (as well as others). The existing purge procedures may be improved by scheduling the purge script to be run less frequently and at times when the network elements reporting to the statistics servers have relatively less statistical data to report. For example, in one embodiment of the invention the purge procedure is scheduled to run once daily early in the morning (e.g., 2 AM local time) because that is a time during which there is relatively less telecommunications

activity, and therefore relatively less statistical data being sent to the statistics servers 103 by the switches and/or other network elements.

[0013]   Another way that existing statistical record purge procedures may be improved is by adding instructions to the purge script that ensure there is adequate log space in the fault servers before the purge script begins deleting records from the statistics database. In most servers there is a dedicated portion of memory (i.e., temporary memory) to which the central processing unit temporarily stores data before writing or sending the data out to a more permanent memory space. For example, the fault servers 103 employed in the assignee of the present invention's network 101 store statistical records in onboard memory and periodically communicate the records that have accumulated in that log space to the statistics database 107.

[0014]   Because the statistical record purge procedure takes a finite amount of time and results in the statistics database being inaccessible to the statistics server(s) while the procedure is running, the temporary memory or log space for the statistics server(s) must have sufficient room to store all the new records generated by the statistics server(s) while the purge procedure is running. Otherwise, potentially important newly generated statistical records may be lost due to log space overflow. In an exemplary embodiment of the invention the additional instructions calculate whether there is 90% or more log space free before calling on the purge script to execute. If not, the scheduler on the fault server waits for approximately 30 seconds before checking the free log space again. The amount of free log space that is required will necessarily depend on a number of factors such as the number of statistical records likely to be generated during the time the purge script is running, how often the purge procedure is run (e.g., hourly, bi-hourly, daily, etc.), the size of the log space (e.g., 2052 Megabytes in the exemplary embodiment noted above), the size of the statistical records generated by the statistics server(s), and the like. Pseudocode for these instructions in accordance with an exemplary embodiment of the invention appear in Appendix A attached hereto. Also, Fig. 3 illustrates in flow diagram form the method outlined above.

[0015]   Note also that in the exemplary embodiment illustrated in Appendix A, statistical records are maintained in the statistics database(s) for 30 days, the purge procedure is run once daily, and it takes approximately 20 minutes for the purge script to execute once it has been called by the scheduler on the fault server(s).

[0016]   Another improvement to existing statistical record purge procedures that may be implemented is to update the pointers and/or indices in the statistics database(s) following the execution of the purge script. Updating the pointers results in notably faster read and write operations in the statistics database(s). Pseudocode for the instructions to update the pointers and indices in accordance with an exemplary embodiment of the invention are attached hereto as Appendix B.

[0017]   While the invention has been described in connection with various exemplary embodiments depicted in the various figures and appendices, it is to be understood that other embodiments may be used or modifications and additions may be made to the described embodiments for performing the same function of the invention without deviating therefrom. The invention therefore should not be limited to any single embodiment, whether depicted herein or not. Rather, the invention should be accorded the full breadth and scope encompassed by the claims appended below.